

What is claimed is:

1. An apparatus for debugging an electronic system, said electronic system including a target microcontroller (MCU) and at least one ROM connected together, comprising:

a debugger unit which debugs said target MCU;

a ROM/RAM emulator/debugger unit, connected to said target MCU and said debugger unit, for emulating said ROM, and including:

a ROM/RAM emulation memory, connected to said emulator/debugger unit, for storing user program codes downloaded from said debugger unit;

an emulator/debugger microcontroller (MCU), coupled to said emulator/debugger unit, for communicating with, and performing requests from, said debugger unit and said target MCU;

a bus mapping unit, coupled to said emulator/debugger MCU; and

a debugger RAM unit, coupled to said emulator/debugger MCU.

2. The apparatus of claim 1, further comprising a debugger service routine ("debugger SR") downloaded from said debugger unit into said ROM/RAM emulation memory with said user program codes from said debugger unit.

3. The apparatus of claim 2, further comprising a communication buffer implemented in said emulation memory, said communication buffer being disposed to store status, request and data by said target MCU and by said emulator/debugger MCU.

4. The apparatus of claim 3, wherein said target MCU executes said debugger SR to:

copy a "loop to itself" instruction to said debugger RAM unit and then jump to said copied instruction to release access of said ROM/RAM emulation memory;

inform said debugger unit upon executing a software breakpoint or upon completing requests from said emulator/debugger MCU; and

parse requests from said emulator/debugger MCU to perform actions, wherein said requests are stored in one of said emulator/debugger RAM unit and said communication buffer.

5. The apparatus of claim 1, wherein said debugger unit is disposed to:
download and upload user program codes to and from the emulator/debugger MCU;
set, delete, enable and disable breakpoints;
writes codes to the ROM/RAM emulation memory;
show and modify registers and memory; and
perform free-run, step-into, step-out and stop test steps.

6. The apparatus of claim 5, wherein said debugger RAM unit is disposed to:
store data for said target MCU to modify its debugging state and data;
store a request for said target MCU;
store debugging status and data of said target MCU for uploading to said debugger unit after said target MCU has finished said request; and
provide program spaces for said target MCU to execute programs in order to release access to said ROM/RAM emulation memory.

7. The apparatus of claim 1, wherein said bus mapping unit maps said debugger RAM unit to a specified address space, which is different from said ROM/RAM emulation memory, to form a continuous and linear addressing space.

8. The apparatus of claim 3, wherein said emulator/debugger MCU passes debugging requests to said target MCU by one of:
a) said emulator/debugger MCU storing requests in said debugger RAM unit;
said emulator/debugger MCU informing said target MCU to perform said requests;

said target MCU executing programs in said ROM/RAM emulation memory; and

b) said emulator/debugger MCU informing said target MCU to perform said requests;

said emulator/debugger MCU informing said target MCU to copy a “loop to itself” instruction to said debugger RAM unit;

said target MCU jumping to said copied instruction to release access to said ROM/RAM emulation memory; and

upon release by said target MCU, said emulator/debugger MCU storing requests in said communication buffer and informing said target MCU to perform said requests.

9. The apparatus of claim 8, wherein said emulator/debugger MCU returns data and status to said debugger unit under one of two conditions:

a) after said target MCU has executed a software breakpoint instruction, after being informed by said target MCU, said emulator/debugger MCU uploads the content of one of said debugger RAM unit or communication buffer to said debugger unit; or

b) after said target MCU has finished a request by said debugger unit, after being informed by said target MCU, said emulator/debugger MCU uploads the content of one of said debugger RAM unit and communication buffer to said debugger unit.

10. An apparatus for debugging an electronic system, comprising:

a target board that includes a target MCU that has a ROM;

a debugger unit for debugging said target MCU;

a ROM/RAM emulator board, connected to said debugger unit, for emulating said ROM of said target MCU, said emulator board including a ROM/RAM emulation memory and an emulator MCU, said ROM/RAM emulation memory being disposed to store user program codes downloaded from said debugger unit, and said emulator MCU being disposed to read and write data from and to said ROM/RAM emulation memory; and

a debugger board, connected to said emulator board and to said target MCU, said debugger board including a debugger MCU for communicating with said debugger unit

and with said target MCU, and for performing requests from said debugger unit and said target MCU, said debugger board further including a bus mapping unit, and a debugger RAM.

11. The apparatus of claim 10, further comprising a debugger service routine (“debugger SR”) downloaded from said debugger unit into said ROM/RAM emulation memory with said user program codes from said debugger unit.

12. The apparatus of claim 11, further comprising a communication buffer implemented in said emulation memory, said communication buffer being disposed to store status, request and data by said target MCU and by said debugger MCU.

13. The apparatus of claim 12, wherein said target MCU executes said debugger SR to:

copy a “loop to itself” instruction to said debugger RAM and then jump to said copied instruction to release access of said ROM/RAM emulation memory;

inform said debugger unit upon executing a software breakpoint or upon completing requests from said debugger MCU; and

parse requests from said debugger MCU to do actions, wherein said requests are stored in one of said debugger RAM and said communication buffer.

14. The apparatus of claim 10, wherein said debugger unit is disposed to:
download and upload user program codes to and from the emulator/debugger MCU;
set, delete, enable and disable breakpoints;
writes codes to the ROM/RAM emulation memory;
show and modify registers and memory; and
perform free-run, step-into, step-out and stop test steps.

15. The apparatus of claim 14, wherein said debugger RAM is disposed to:

store data for said target MCU to modify its debugging state and data;
store a request for said target MCU;
store debugging status and data of said target MCU for uploading to said debugger unit after said target MCU has finished said request; and
provide program spaces for said target MCU to execute programs in order to release access to said ROM/RAM emulation memory.

16. The apparatus of claim 10, wherein said bus mapping unit maps said debugger RAM to a specified address space, which is different from said ROM/RAM emulation memory, to form a continuous and linear addressing space.

17. The apparatus of claim 12, wherein said debugger MCU passes debugging requests to said target MCU by one of:

- a) said debugger MCU storing requests in said debugger RAM;
said debugger MCU informing said target MCU to perform said requests;
said target MCU executing programs in said ROM/RAM emulation memory; and
- b) said debugger MCU informing said target MCU to perform said requests;
said debugger MCU informing said target MCU to copy a "loop to itself" instruction to said debugger RAM;
said target MCU jumping to said copied instruction to release access to said ROM/RAM emulation memory; and
upon release by said target MCU, said debugger MCU storing requests in said communication buffer and informing said target MCU to perform said requests.

18. The apparatus of claim 17, wherein said debugger MCU returns data and status to said debugger unit under one of two conditions:

- a) after said target MCU has executed a software breakpoint instruction, after being informed by said target MCU, said debugger MCU uploads content of one of said debugger RAM unit or said communication buffer to said debugger unit; or

b) after said target MCU has finished a request by said debugger unit, after being informed by said target MCU, said debugger MCU uploads content of one of said debugger RAM and said communication buffer to said debugger unit.

19. An apparatus for debugging an electronic system, said electronic system including a target micro-processing unit ("target MCU") and at least one ROM connected together, comprising:

a debugger unit for debugging said target MCU;

a debugger MCU, implemented with said target MCU, for communicating with said debugger unit and with said target MCU, and for performing requests from said debugger unit and said target MCU;

a ROM/RAM emulator board, connected to said debugger unit, for emulating said ROM of said target MCU, said emulator board including a ROM/RAM emulation memory and an emulator MCU, said ROM/RAM emulation memory being disposed to store user program codes downloaded from said debugger unit, and said emulator MCU being disposed to read and write data from and to said ROM/RAM emulation memory; and

a debugger RAM board, connected to said emulator board and to said target MCU, said debugger RAM board including a bus mapping unit, and a debugger RAM.

20. The apparatus of claim 19, further comprising a debugger service routine ("debugger SR") downloaded from said debugger unit into said ROM/RAM emulation memory with said user program codes from said debugger unit.

21. The apparatus of claim 20, further comprising a communication buffer implemented in said emulation memory, said communication buffer being disposed to store status, request and data by said target MCU and by said debugger MCU.

22. The apparatus of claim 19, further including a target/debugger board, connected to said debugger RAM board, that includes said target MCU and said debugger MCU.

23. An apparatus for debugging an electronic system, said electronic system including a target microcontroller (MCU) and at least one ROM connected together, comprising:

a debugger unit for debugging said target MCU;

a debugger MCU, implemented with said target MCU, for communicating with said debugger unit and with said target MCU, and for performing requests from said debugger unit and said target MCU;

a ROM/RAM emulator board, connected to said debugger unit, said emulator board being disposed to emulate said ROM of said target MCU, said emulator board including a ROM/RAM emulation memory and an emulator MCU, said emulation memory being disposed to store user program codes downloaded from said debugger unit, said emulator MCU being disposed to read and write data from and to said ROM/RAM emulation memory; and

a target/debugger board, connected to said emulator board, said target/debugger board including a bus mapping unit, a debugger RAM, said target MCU and said debugger MCU.

24. The apparatus of claim 23, further comprising a debugger service routine ("debugger SR") downloaded from said debugger unit into said ROM/RAM emulation memory with said user program codes from said debugger unit.

25. The apparatus of claim 24, further comprising a communication buffer implemented in said emulation memory, said communication buffer being disposed to store status, request and data by said target MCU and by said debugger MCU.

26. An apparatus for debugging an electronic system, said electronic system including a micro-controller ("target MCU") and at least one ROM connected together, comprising:

a debugger unit for debugging said target MCU;

a ROM/RAM emulator board, connected to said debugger unit, said emulator board being disposed to emulate said ROM of said target MCU, said emulator board including a ROM/RAM emulation memory and an emulator MCU, said emulation memory being

disposed to store user program codes downloaded from said debugger unit, said emulator MCU being disposed to read and write data from and to said ROM/RAM emulation memory;

a debugger MCU, implemented with said target MCU, for communicating with said debugger unit and with said target MCU, and for performing requests from said debugger unit and said target MCU, said debugger MCU including an embedded RAM;

a target/debugger board, connected to said emulator board and to said debugger unit, said target/debugger board including said target MCU, said debugger MCU, and an embedded debugger RAM.

27. The apparatus of claim 26, further comprising a debugger service routine ("debugger SR") downloaded from said debugger unit into said ROM/RAM emulation memory with said user program codes from said debugger unit.

28. The apparatus of claim 27, further comprising a communication buffer implemented in said emulation memory, said communication buffer being disposed to store status, request and data by said target MCU and by said debugger MCU.

29. An apparatus for debugging an electronic system, comprising:

a debugger unit;

a ROM/RAM emulator/debugger/target board, connected to said debugger unit, said emulator/debugger/target board including:

an emulator/debugger/target MCU having a target MCU that is debugged by the debugger unit, the emulator/debugger/target MCU communicating with said debugger unit, and performing requests from said debugger unit and said target MCU;

a debugger RAM;

a bus mapping unit;

a ROM memory, disposed to store service routines; and

a ROM/RAM emulation memory being disposed to store user program codes downloaded from said debugger unit.

30. The apparatus of claim 29, further comprising a debugger service routine (“debugger SR”) which is stored in the ROM memory.

31. The apparatus of claim 29, wherein said debugger RAM is implemented inside the emulator/debugger/target MCU.

32. An apparatus for debugging an electronic system, comprising:

a debugger unit;

a ROM/RAM emulator/debugger/target board, connected to said debugger unit, said emulator/debugger/target board including:

an emulator/debugger/target MCU having a target MCU that is debugged by the debugger unit, the emulator/debugger/target MCU communicating with said debugger unit, and performing requests from said debugger unit and said target MCU, said emulator/debugger/target MCU including a debugger RAM and a ROM memory;

a ROM/RAM emulation memory being disposed to store user program codes downloaded from said debugger unit.